

Praktikumsausarbeitung

im Studiengang

Allgemeine Informatik

IPv6

Internet Protocol Version 6

Referent:	Tutor des Praktikums
Korreferent:	Tutor des Praktikums
Vorgelegt am:	Redacted
Vorgelegt von:	Jana Zimmer
	Redacted
	Robert-Gerwig-Platz 1
	78120 Furtwangen
	Redacted

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

Abkürzungsverzeichnis

DAD . . . . Duplicate Address Detection
DNS . . . . Domain Naming System

IPv4 . . . . Internet Protocol Version 4
IPv6 . . . . Internet Protocol version 6
ISP . . . . Internet Service Provider

ND . . . . Neighbor Discovery
NDP . . . . Neighbor Discovery Protocol

QoS . . . . Quality of Service

RIPE NCC . . . . Réseaux IP Européens Network Coordination Centre
RIR . . . . Regional Internet Registries
RR . . . . Ressource Record

SLAAC . . . . Stateless Address (Auto) Configuration

Listingverzeichnis

1 Recherche

1.1 Grundlegender Aufbau von IPv6

1.1.1 Header

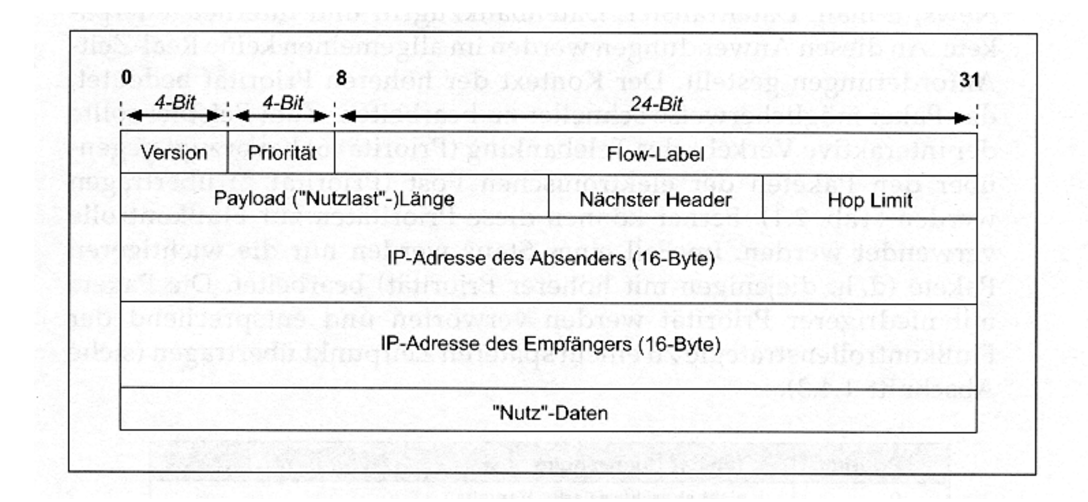


Abbildung 1.1: IPv6 Header
Abbildung von Poiger (Poiger, 1998)

Im Vergleich zum Header bei Internet Protocol Version 4 (IPv4) (siehe Abb. A.1) hat der Header bei IPv6 weniger Felder. Die Felder für die Adresse des Empfängers und die Adresse des Senders bleiben natürlich bestehen, auch wenn ihre Größe geändert wurde (eine IPv6-Adresse hat 128 Bit und eine IPv4-Adresse hat 32 Bit).
Das Feld für Version enthält hier natürlich die 6 (Deering und Hinden, 1995, p. 5) um das Paket als Paket des Internet Protocol version 6 zu kennzeichnen.

Priorität Das Prioritätsfeld enthält eine 4 Bit Zahl (Deering und Hinden, 1995, p. 5) welche es dem Sender erlaubt der Zustellung des Pakets eine gewisse Priorität zuzuteilen. Die Werte 0 bis 7 (siehe Tab. 1.1) sind für Traffic für welche der Sender über *congestion control* verfügt, z.B. TCP-Traffic. Die Werte 8 bis 15 sind für die über keine Flusskontrolle verfügen, bsp. real-time Pakete welche konstant versendet werden. Für Traffic ohne Flusskontrolle stellt die Priorität 8 die niedrigste Priorität (als Einsatzbeispiel nennt Deering und Hinden high-fidelity video traffic) und Priorität 15 die höchste Priorität (als Einsatzbeispiel nennt Deering und Hinden low-fidelity audio traffic) dar.

0	uncharacterized traffic
1	"filler" traffic (e.g., netnews)
2	unattended data transfer (e.g., email)
3	(reserved)
4	attended bulk transfer (e.g., FTP, NFS)
5	reserved
6	interactive traffic (e.g., telnet, X)
7	internet control traffic (e.g., routing protocols, SNMP)

Tabelle 1.1: Prioritäten aus RFC 1883
Wortlaut unverändert, Format angepasst (Deering und Hinden, 1995)

Flow Label Dieses Feld enthält ein Flow-Label der Länge 24 Bit (Deering und Hinden, 1995, p. 5). Das Flow-Label kann dazu genutzt werden um *Spezialbehandlung* für Pakete von Routern zu erbeten (Deering und Hinden, 1995, p. 28). Deering und Hinden zählen hier als Beispiele nicht standardmäßigen Quality of Service (QoS) (»non-default quality of service«) oder "real time" service (Deering und Hinden, 1995, p. 28). Hosts oder Router die das nicht unterstützen müssen das Feld wenn sie der Ursprung das Paketes sind auf null setzen, wenn sie das paket weiterleiten, das Feld unverändert weiterreichen und wenn sie Empfänger des Paketes sind ignorieren (Deering und Hinden, 1995, p. 28).

> »A flow is a sequence of packets sent from a particular source to a particular (unicast or multicast) destination for which the source desires special handling by the intervening routers. The nature of that special handling might be conveyed to the routers by a control protocol, such as a resource reservation protocol, or by information within the flow's packets themselves, e.g., in a hop-by-hop option. « (Deering und Hinden, 1995, p. 28)

1.1.2 Protokolle

Upper-Layer Protocols Für Protokolle in höheren Schichten weist RFC 1883 darauf hin, dass wenn die IP-Adresse Teil der Prüfsumme des Paketes auf höherer Schicht ist, muss das Protokoll so angepasst werden, dass die Prüfsumme für IPv6-Adressen generiert wird und nicht IPv4-Adressen. In Abbildung 1.2 ist ein Pseudoheader zu erkennen, wie er in RFC 1883 beschrieben ist. Im Feld Next-Header wird angegeben welches Upper-Layer Protokoll folgt (UDP: 17, TCP: 6).

Domain Naming System (DNS) Domain Naming System (DNS) funktioniert grundlegend nach dem selben Prinzip wie bei IPv4. Es wird einfach ein anderer RR-Typ (AAAA) verwendet (Ksinant et al., 2003, p. 3). Während bei IPv4 für IP-Adressen in DNS-Records der RR-Typ A verwendet wird.

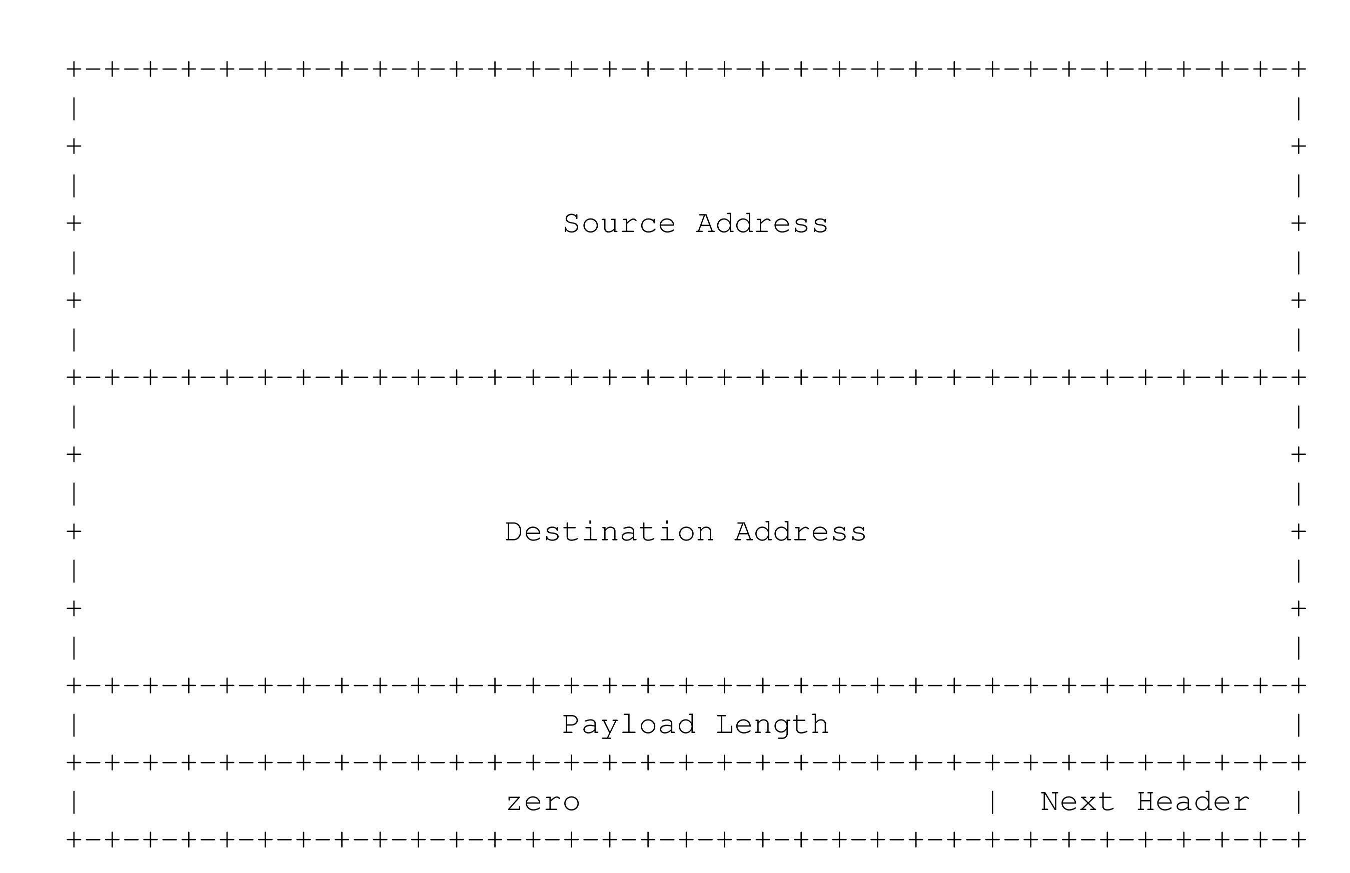
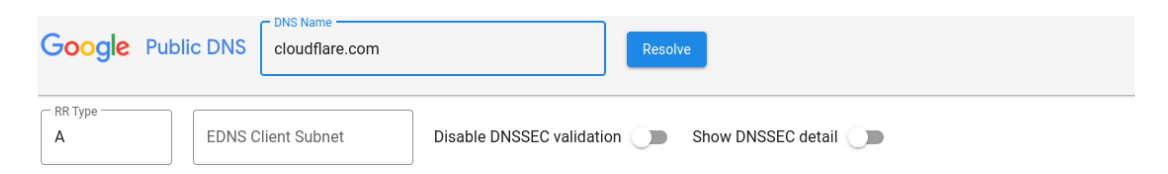


Abbildung 1.2: Pseudo-Header für TCP/UDP über IPv6
Abbildung von Deering und Hinden (Deering und Hinden, 1995)



Result for cloudflare.com/A with DNSSEC validation and without DNSSEC detail:


```
{
  "Status": 0 /* NOERROR */,
  "TC": false,
  "RD": true,
  "RA": true,
  "AD": true,
  "CD": false,
  "Question": [
    {
      "name": "cloudflare.com.",
      "type": 1 /* A */
    }
  ],
  "Answer": [
    {
      "name": "cloudflare.com.",
      "type": 1 /* A */,
      "TTL": 300,
      "data": "104.16.133.229"
    },
    {
      "name": "cloudflare.com.",
      "type": 1 /* A */,
      "TTL": 300,
      "data": "104.16.132.229"
    }
  ],
  "Comment": "Response from 162.159.5.6."
}
```


Abbildung 1.3: A for dns cloudflare.com

1.2 Vergabe von IP-Adressen

1.2.1 Vergabe von IPv4-Adressen

Die Vergabe der IP-Adressen ist regional durch RIRs organisiert. In Europa und dem sogenannten Mittleren Osten ist dafür beispielsweise die Réseaux IP Européens Network Coordination Centre (RIPE NCC) zuständig.

Jedoch erfolgte die Vergabe der IP-Adressbereiche so, dass manche Regionen der Welt welche besonders bevölkerungsstark sind nicht besonders viele IP-Adressen zur Verfügung haben.

Die RIR können dann ISPs IP Adressräume zur Verfügung stellen, aus welchen dann die Endnutzer IP-Adressen erhalten, mit welchen Sie aufs Internet zugreifen können.

1.2.2 Vergabe von IPv6-Adressen

Die IPv6 kann bsp. mit Stateless Address (Auto) Configuration (SLAAC) (siehe Kap. 2.2.1) oder DHCPv6 (siehe Kap. 2.2.3) vergeben werden.

Google Public DNS
DNS Name
cloudflare.com
Resolve

RR Type
AAAA
EDNS Client Subnet
Disable DNSSEC validation
Show DNSSEC detail

Result for cloudflare.com/AAAA with DNSSEC validation and without DNSSEC detail:


```
{
  "Status": 0 /* NOERROR */,
  "TC": false,
  "RD": true,
  "RA": true,
  "AD": true,
  "CD": false,
  "Question": [
    {
      "name": "cloudflare.com.",
      "type": 28 /* AAAA */
    }
  ],
  "Answer": [
    {
      "name": "cloudflare.com.",
      "type": 28 /* AAAA */,
      "TTL": 300,
      "data": "2606:4700::6810:85e5"
    },
    {
      "name": "cloudflare.com.",
      "type": 28 /* AAAA */,
      "TTL": 300,
      "data": "2606:4700::6810:84e5"
    }
  ],
  "Comment": "Response from 162.159.8.55."
}
```


Abbildung 1.4: AAAA for dns cloudflare.com

1.3 Grund um Schrittweise auf IPv6 umzusteigen

Der Hauptgrund um auf IPv6 umzusteigen mag wohl sein, dass die Anzahl der IPv4-Adressen in manchen Weltregionen einfach zu gering sind, für die Zahl der Netzteilnehmer. Da mit der 4mal solangen IP-Adresse bei IPv6 weit mehr Netzteilnehmer am Netz teilnehmen können, oder anders formuliert es können viel mehr Geräte eine global eindeutige IP-Adresse bekommen. Während es bei IPv4 maximal *nur* rund 4 Milliarden Adressen geben kann, kann es bei IPv6 bis zu ungefähr 340 Sextillionen Adressen geben. "Bildlich gesprochen: Jedem Quadratmillimeter der Erde inklusive Ozeane stehen dann 600 Billiarden Adressen zu!"(Lehrerfortbildungsserver, 2016).

Warum Schrittweise? Es gibt immer noch Systeme/Dienste welche nicht (gut) mit IPv6 funktionieren. Weshalb eine schlagartige Umstellung auf IPv6 wohl ziemlich verheerend für diese wären.n Und da IPv4 Infrastruktur nicht direkt mit IPv6-Infrastruktur interoperabel (Wikipedia contributors, 2022) ist, mussten Übergangstechnologien geschaffen werden.



Abbildung 1.5: Regional Internet Registries (RIR) World Service Regions
Abbildung von Wikipedia contributors, 2021

1.3.1 Übergangstechnologien

Um den Übergang von IPv4 zu IPv6 zu ermöglichen gibt es zahlreiche Technologien, wie beispielsweise Dual-Stack Lite (Wikipedia contributors, 2022).

Dual IP Layer Operation Die *most straightforward* Methode um eine Interoperabilität von IPv4 und IPv6 herzustellen ist, dass alle IPv6-Netzknoten weiterhin mit IP4 kompatibel sind, also IPv4 komplett implementieren (Gilligan und Nordmark, 2005, p. 4). Die Netzknoten, die so verfahren können direkt mit IPv4 und IPv6 umgehen, und können sowohl IPv4 als auch IPv6 verwenden und entsprechende Netzpakete empfangen und versenden (Gilligan und Nordmark, 2005, p. 4).

1.4 Neighbor Discovery Protocol (NDP)

Das Neighbor Discovery Protocol (NDP) gilt als Ersatz für das ARP-Protokoll bei IPv4. Es dient dazu um Nachbarn schnell zu erkennen und deren Link-Layer-Adresse zu bestimmen und gecachte Adressen, welche nicht mehr gültig sind rasch zu löschen (Simpson et al., 2007, p. 4).
Also wird es dazu genutzt um zu detektieren, welche Nachbarn erreichbar sind und welche nicht, bzw. geänderte Link-Layer-Adressen festzustellen (Simpson et al., 2007, p. 4). Wenn ein Pfad zu einem Host oder ein Pfad zu einem Router versagt, sucht dieses Protokol alternativ nach alternativen Pfaden(Simpson et al., 2007, p. 4).
Um das zu bewerkstelligen verwended Neighbor Discovery (ND) für manche Services Link-Layer-Multicast, welche bei manchen Link-Typen womöglich nicht verwendbar sind.
Simpson et al. beschreibt Neighbor als Netzknoten beschrieben die an den selben *link*

angeschlossen sind (Simpson et al., 2007, p. 5). Und Link wird wie folgt beschrieben: ä communication facility or medium over which nodes can communicate at the link layer, i.e., the layer immediately below IP. Examples are Ethernets (simple or bridged), PPP links, X.25, Frame Relay, or ATM networks as well as Internet-layer (or higher-layer) "tunnels", such as tunnels over IPv4 or IPv6 itself."(Simpson et al., 2007, p. 5).

1.5 Router Advertisments

Router teilt Clients bsp. den IPv6-Präfix mit (welche bsp. für Stateless Address (Auto) Configuration (SLAAC) verwendet wird).

1.6 Mobile IPv6

Mobile IPv6 bezeichnet einen Ansatz bei welcher ein Client unter der selben IPv6-Adressse erreichbar ist, egal wo sich dieser aufhält (vgl. Johnson et al., 2011, abstract). Wenn der Client gerade nicht *zuhause* ist verfügt er auch über eine Care-Of-Address welche vom jeweiligen Netzanschluss abhängig ist (vgl. Johnson et al., 2011, abstract). Wenn Pakete an die Home-Address gesendet werden, werden diese transparent für den Sender an die Care-Of-Address geroutet (vgl. Johnson et al., 2011, abstract), wobei ein Client über mehrere Care-Of-Addresses verfügen kann (vgl. Johnson et al., 2011, p. 15). Um Daten an den Client zu senden, wenn dieser sich nicht zuhause befindet müssen Daten hierbei nicht über den sog. Home-Agent geleitet werden, sondern können direkt an den Client gesendet werden (vgl. Johnson et al., 2011, abstract). Somit wird erreicht, dass ein Client immer über die Home-Address erreichbar ist. Die Home-Address ist eine IP-Adresse aus dem zur verfügung gestellten IP-Adressbereich (vgl. Johnson et al., 2011, p. 15).

1.7 ICMPv6

Nach Davies und Mohácsi ist ICMPv6 essentiell für die Nutzung von IPv6, wenn doch gleich die unkontrollierte Weiterleitung von ICMPv6-Nachrichten einen möglichen Sicherheitsgefahr darstellt (vgl. Davies und Mohácsi, 2007). Bei diesem Protokoll wird in das Next-Header-Feld vom IPv6-Header die Nummer 58 stehen.

2 Schritt 2

2.1 bezogene IPv6-Adresse

Abbildung 2.1: ipconfig /all unter Windows

2.1.1 unter Windows

IPv6-Adressen:

- fd97:[redacted]:cc98:5c0:fbb8(Preferred)
- fd97:[redacted]:b0f7:105:6d60(Preferred)
- fe80::fc21:cc98:5c0:fbb8%7(Preferred)

MAC-Adresse: [redacted]

2.1.2 unter Linux

IPv6-Adressen:

- 2001:[redacted]:6ac4:4d6b/64 scope global temporary dynamic
- 2001:[redacted]:feb1:6f20/64 scope global dynamic mngtmpaddr noprefixroute

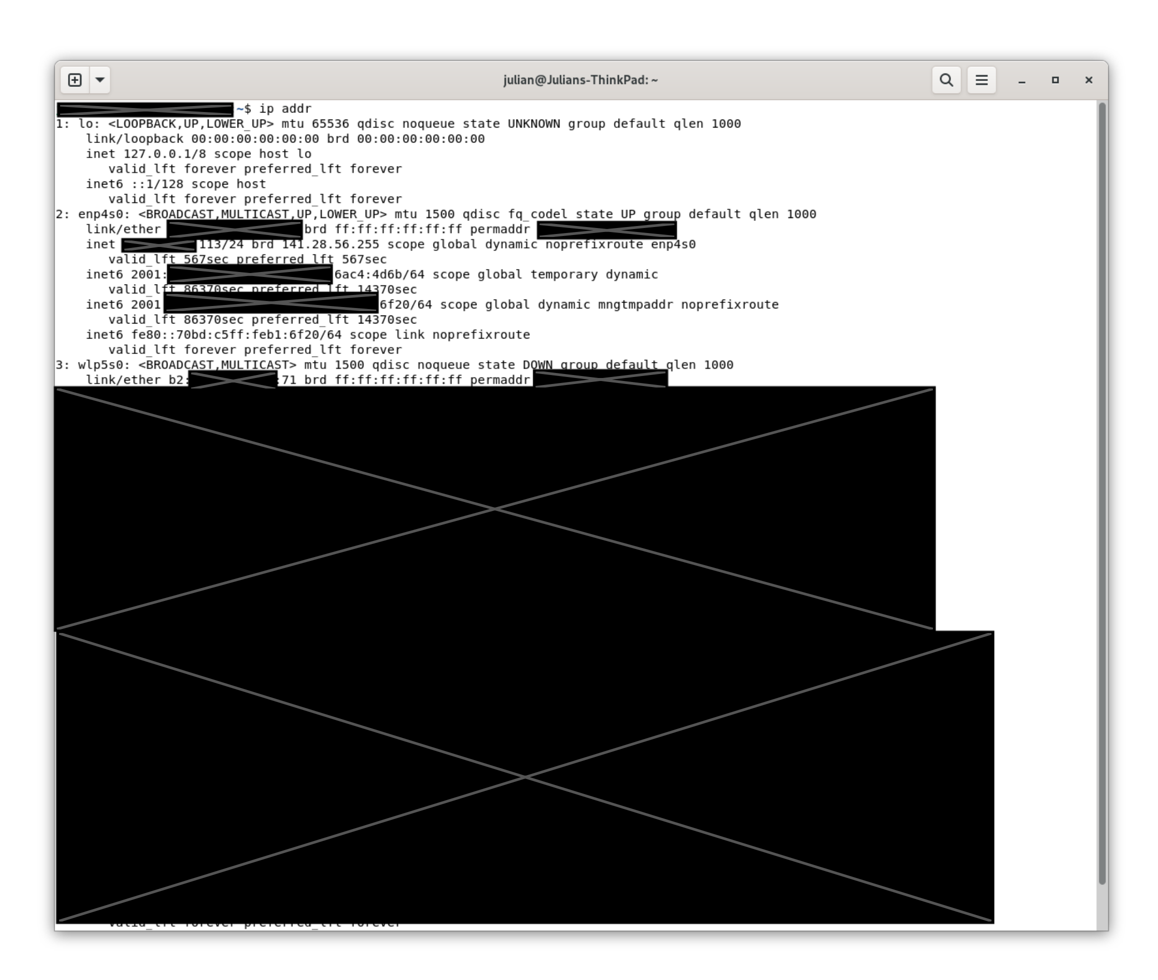


Abbildung 2.2: IP unter Linux

- fe80::[redacted]:6f20/64 scope link noprefixroute

MAC-Adresse: [redacted]

Aufälligkeiten Bei der dritten IPv6-Adresse fällt auf, dass Teile der IPv6-Adresse mit der MAC-Adresse Ähnlichkeit haben, so kommt die Sequenz bdc5 und b16f20 in beiden Adressen vor. Dies scheint allerdings nur beim Linux-Rechner der Fall zu sein (vgl. Abb. 2.2), bei Windows ist dies nicht beobachtbar (vgl. Abb. 2.1).

2.2 Adressvergabe bei IPv6

2.2.1 Stateless Address (Auto) Configuration (SLAAC)

Bei Stateless Address (Auto) Configuration (SLAAC) wird die IPv6-Adresse vom Netzteilnehmer selbst erzeugt. Es werden zwischen IPv6-Adressen mit *local Scope* und IPv6-Adressen mit *global Scope* unterschieden (wie auch bei der Ausgabe erkennbar ist: Abb. 2.2). Mithilfe von SLAAC können beide erzeugt werden. Nach Kompendium, n. d. bietet dies den selben Kompfort wie ein einfach gehaltener DHCP-Server. (Kompendium, n. d.) Wie in Abb. 2.3 zu erkennen wird, kann die IPv6-Adresse mithilfe von

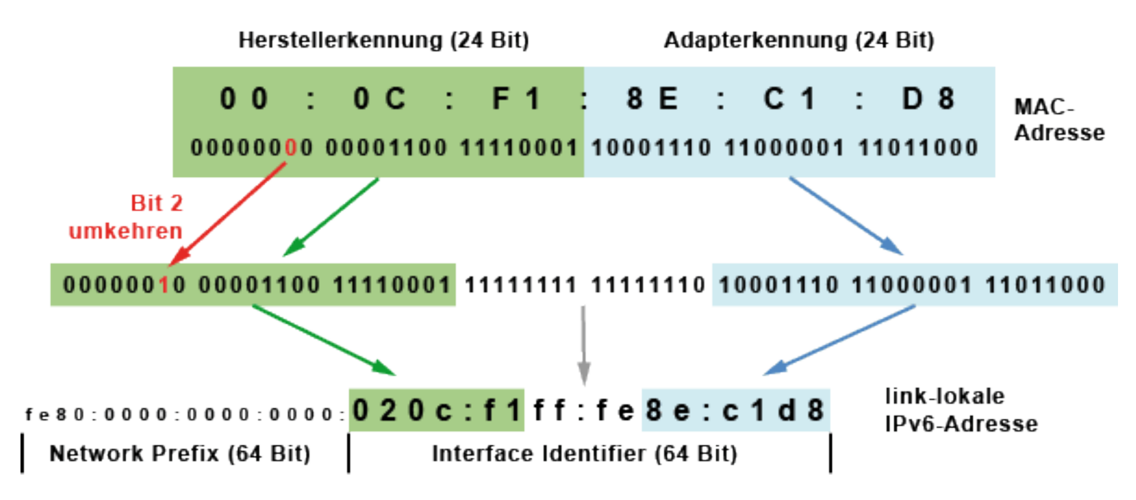


Abbildung 2.3: SLAAC für eine link-lokale IPv6-Adresse

SLAAC aus der MAC-Adresse (bzw. EUI-64-Identifier) gebildet werden.

> In der Mitte der 48-Bit-MAC-Adresse (zwischen dem dritten und dem vierten Byte) werden mit „ff:fe"zwei feste Bytes eingefügt, damit es 64 Bit werden. Zusätzlich wird noch das zweite Bit im ersten Byte der MAC-Adresse invertiert. Das heißt, aus „1"wird „0"und aus „0"wird „1".
> Kompendium, n. d.

2.2.2 Duplicate Address Detection (DAD)

Mithilfe von Duplicate Address Detection (DAD) werden doppelte Adressen vermieden. Dazu wird ein Neighbor Solicitation und Neighbor Advertisment durchgeführt. Bei Neighbor Solicitation wird eine Anfrage an die generierte Adresse ins lokale Netz gesendet, als Antwortadresse dient eine Multicast-Adresse, es sollte keiner antworten, wenn jemand antwortet (Neighbor Advertisment) , ist die Adresse bereits vergeben.7

```
▸ Frame 97: 80 bytes on wire (640 bits), 80 bytes captured (640 bits) on interface any, id 0
▸ Linux cooked capture v1
▸ Internet Protocol Version 6, Src: fe80::6efe:54ff:fe02:aee1, Dst: fe80::64fb:17ff:fe9e:1581
▾ Internet Control Message Protocol v6
    Type: Neighbor Advertisement (136)
    Code: 0
    Checksum: 0x6194 [correct]
    [Checksum Status: Good]
  ▾ Flags: 0xc0000000, Router, Solicited
      1... .... .... .... .... .... .... .... = Router: Set
      .1.. .... .... .... .... .... .... .... = Solicited: Set
      ..0. .... .... .... .... .... .... .... = Override: Not set
      ...0 0000 0000 0000 0000 0000 0000 0000 = Reserved: 0
    Target Address: 2001:7c0:f00:2056::2
```

Abbildung 2.4: Beispiels Neighbor Advertisement

2.2.3 DHCPv6

Analog zu DHCPv4 (für IPv4) gibt es DHCPv6 für IPv6, welche als Stateful Address Configuration klassifiziert wird. Auch wenn DHCP für IPv6 dank SLAAC eigentlich

```
▸ Frame 96: 88 bytes on wire (704 bits), 88 bytes captured (704 bits) on interface any, id 0
▸ Linux cooked capture v1
▸ Internet Protocol Version 6, Src: [redacted], Dst: [redacted]
▾ Internet Control Message Protocol v6
    Type: Neighbor Solicitation (135)
    Code: 0
    Checksum: 0xa5bb [correct]
    [Checksum Status: Good]
    Reserved: 00000000
    Target Address: 2001:7c0:f00:2056::2
  ▾ ICMPv6 Option (Source link-layer address : [redacted]
      Type: Source link-layer address (1)
      Length: 1 (8 bytes)
      Link-layer address: 66:fb:17:9e:15:81 (66:fb:17:9e:15:81)
```

Abbildung 2.5: Beispiels Neighbor Solicitation

nicht notwendig ist, was für kleine Netzwerker auch gut geeignet ist, für große Netzwerke ist dies aber keine gute Idee, weil dies die Verwaltung erschwert.

Ablauf Als erstes generiert sich der Host selbst eine IPv6-Adresse, welche bis zum nächsten Router gültig ist. Der Host empfängt dann ein Router Advertisement mit dem Flag „managed“ vom nächsten Router, somit weiß der Host dann, dass er DHCP verwenden soll. Schließlich durchläuft der Host die Kommunikation mit dem DHCPv6-Server und erhält von ihm eine vollständige IPv6-Konfiguration (mit globaler Adresse, DNS-Server unter weitere netzspezifische Adressen und Parameter).

3 Schritt 3

Web-Adressen welche nur via IPv4 erreichbar sind, haben meist kein AAAA-Eintrag in ihrmen DNS-Record. Wie am Beispiel der Domain gar-vs.de (bzw. hs-furtwangen.de) demonstriert verfügt, der DNS-Record zwar über ein A-Eintrag (siehe Abbildung B.1, bzw. Abbildung B.3) verfügt also über eine IPv4-Adresse, aber über kein AAAA-Eintrag (siehe Abbildung B.2, bzw. Abbildung B.4), verfügt also über keine IPv6-Adresse und ist somit nicht über ausschließlich IPv6 erreichbar. Anhand der Domain cloudflare.com wurde demonstriert, welche aus dem IPv6-Netz erreichbar ist, das dieser AAAA-Record (siehe Abb. B.6) notwendig ist, für die Erreichbarkeit aus dem IPv6-Netz. Weiterhin, zeigt der DNS-Record (sowie A (Abb. B.5) und AAAA (Abb. B.6)), dass zu einer Domain, auch mehrere IP-Adressen hinterlegt werden können.

4 Schritt 4

```
[redacted]~$ ping6 cloudflare.com
PING cloudflare.com(2606:4700::6810:84e5 (2606:4700::6810:84e5)) 56 data bytes
64 bytes from 2606:4700::6810:84e5 (2606:4700::6810:84e5): icmp_seq=1 ttl=54 time=9.36 ms
64 bytes from 2606:4700::6810:84e5 (2606:4700::6810:84e5): icmp_seq=2 ttl=54 time=9.29 ms
64 bytes from 2606:4700::6810:84e5 (2606:4700::6810:84e5): icmp_seq=3 ttl=54 time=8.38 ms
64 bytes from 2606:4700::6810:84e5 (2606:4700::6810:84e5): icmp_seq=4 ttl=54 time=8.38 ms
64 bytes from 2606:4700::6810:84e5 (2606:4700::6810:84e5): icmp_seq=5 ttl=54 time=9.30 ms
64 bytes from 2606:4700::6810:84e5 (2606:4700::6810:84e5): icmp_seq=6 ttl=54 time=9.45 ms
64 bytes from 2606:4700::6810:84e5 (2606:4700::6810:84e5): icmp_seq=7 ttl=54 time=8.36 ms
64 bytes from 2606:4700::6810:84e5 (2606:4700::6810:84e5): icmp_seq=8 ttl=54 time=8.38 ms
64 bytes from 2606:4700::6810:84e5 (2606:4700::6810:84e5): icmp_seq=9 ttl=54 time=7.37 ms
^C
--- cloudflare.com ping statistics ---
9 packets transmitted, 9 received, 0% packet loss, time 8013ms
rtt min/avg/max/mdev = 7.369/8.696/9.449/0.659 ms
[redacted]~$
```

Abbildung 4.1: ping6 clouflare.com

```
[redacted]~$ ping6 2001:7c0:f00:2056:180f:5132:522c:45c0
PING 2001:7c0:f00:2056:180f:5132:522c:45c0(2001:7c0:f00:2056:180f:5132:522c:45c0) 56 data bytes
64 bytes from 2001:7c0:f00:2056:180f:5132:522c:45c0: icmp_seq=1 ttl=64 time=0.881 ms
64 bytes from 2001:7c0:f00:2056:180f:5132:522c:45c0: icmp_seq=2 ttl=64 time=0.922 ms
64 bytes from 2001:7c0:f00:2056:180f:5132:522c:45c0: icmp_seq=3 ttl=64 time=1.06 ms
64 bytes from 2001:7c0:f00:2056:180f:5132:522c:45c0: icmp_seq=4 ttl=64 time=0.870 ms
64 bytes from 2001:7c0:f00:2056:180f:5132:522c:45c0: icmp_seq=5 ttl=64 time=1.11 ms
64 bytes from 2001:7c0:f00:2056:180f:5132:522c:45c0: icmp_seq=6 ttl=64 time=1.11 ms
64 bytes from 2001:7c0:f00:2056:180f:5132:522c:45c0: icmp_seq=7 ttl=64 time=1.04 ms
64 bytes from 2001:7c0:f00:2056:180f:5132:522c:45c0: icmp_seq=8 ttl=64 time=0.974 ms
^C
--- 2001:7c0:f00:2056:180f:5132:522c:45c0 ping statistics ---
8 packets transmitted, 8 received, 0% packet loss, time 7011ms
rtt min/avg/max/mdev = 0.870/0.996/1.113/0.092 ms
```

Abbildung 4.2: ping6 anderer PC

4.1 Wireshark

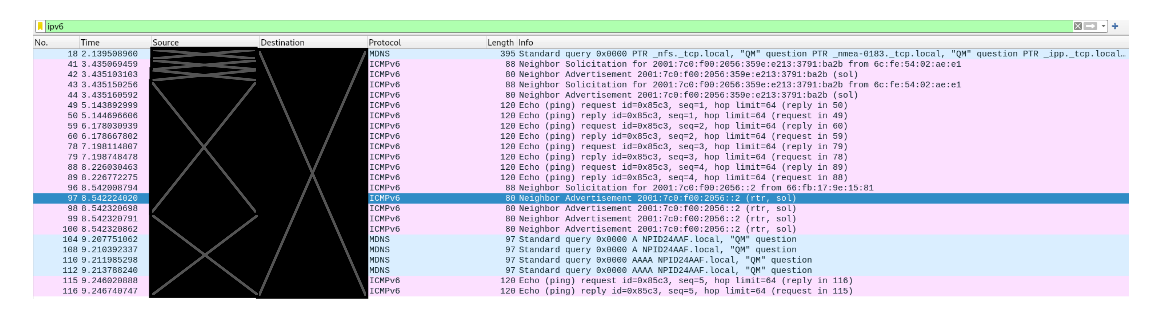


Abbildung 4.3: Wireshark von IPv6-Verkehr

4.2 traceroute6

Es fällt auf, dass Traceroute für PC1 nach PC2 2 Hops zeigt (siehe Abb. 4.4) und von PC2 nach PC1 nur 1 Hop (siehe Abb. 4.5). Logischerweise zeigt ein Traceroute zu einem Host im Internet (bsp. cloudflare.com, siehe Abb. 4.6) mehr Hops wie ein Traceroute zu einem Host im selben lokalen Netz.

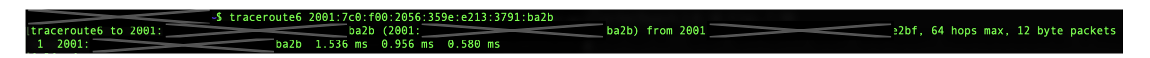


Abbildung 4.4: Traceroute von PC1 nach PC2

```
traceroute to 2001:[redacted]45c0 (2001:[redacted]:45c0), 30 hops max, 80 byte packets
 1  2001:7c0:f00:2056::2 (2001:7c0:f00:2056::2)  0.363 ms  0.322 ms  0.317 ms
 2  2001:[redacted]:45c0 (2001:[redacted]:45c0)  1.148 ms  1.224 ms  1.209 ms
```

Abbildung 4.5: Traceroute von PC2 nach PC1

```
traceroute to cloudflare.com (2606:4700::6810:84e5), 30 hops max, 80 byte packets
 1  2001:7c0:f00:2056::2 (2001:7c0:f00:2056::2)  0.333 ms  0.307 ms  0.291 ms
 2  2001:7c0:f00:1fff:ffff:ffff:ffff:fe0d (2001:7c0:f00:1fff:ffff:ffff:ffff:fe0d)  0.900 ms  1.007 ms  1.113 ms
 3  * * *
 4  * * *
 5  * * *
 6  kar-rz-a99-hu0-3-0-5.belwue.net (2001:7c0:2:109e::)  5.483 ms  5.438 ms  5.461 ms
 7  * * stu-nwz-a99-hu0-3-0-0.belwue.net (2001:7c0:2:10f4::1)  7.514 ms
 8  * * *
 9  as13335.frankfurt.megaport.com (2001:7f8:8:20:0:3417:0:1)  57.073 ms  49.576 ms  9.951 ms
10  2400:cb00:471:3:: (2400:cb00:471:3::)  10.165 ms  7.889 ms  8.994 ms
11  2400:cb00:71:1024::a29e:5582 (2400:cb00:71:1024::a29e:5582)  7.299 ms 2400:cb00:470:1024::ac46:f122 (2400:c
b00:470:1024::ac46:f122)  10.509 ms 2400:cb00:471:1024::ac46:f51a (2400:cb00:471:1024::ac46:f51a)  9.342 ms
```



Literaturverzeichnis

Davies, E. B. & Mohácsi, J. (2007). Recommendations for Filtering ICMPv6 Messages in Firewalls. https://doi.org/10.17487/RFC4890

Deering, D. S. E. & Hinden, B. (1995). Internet Protocol, Version 6 (IPv6) Specification. https://doi.org/10.17487/RFC1883

Gilligan, R. E. & Nordmark, E. (2005). Basic Transition Mechanisms for IPv6 Hosts and Routers [This document obsoletes RFC 2893; RFC Standards Track]. https://doi.org/10.17487/RFC4213

Johnson, D. B., Arkko, J. & Perkins, C. E. (2011). Mobility Support in IPv6. https://doi.org/10.17487/RFC6275

Kompendium, E. (n. d.). SLAAC - Stateless Address Autoconfiguration (IPv6). https://www.elektronik-kompendium.de/sites/net/1902131.htm

Ksinant, V., Huitema, C., Thomson, D. S. & Souissi, M. (2003). DNS Extensions to Support IP Version 6 [This document obsoletes RFC 3152 and RFC 1886; RFC Standards Track]. https://doi.org/10.17487/RFC3596

Lehrerfortbildungsserver. (2016). Adressierung. https://lehrerfortbildung-bw.de/u_matnatech/imp/gym/bp2016/fb1/3_i3_run/1_hintergrund/3_infos/3_adresse/

Poiger, J. (1998). IPv6 Header und Sicherheitskonzept [(Accessed on 06/15/2022)]. https://www2.hs-fulda.de/~klingebiel/nbs-kolloquium/ipng2/

Simpson, W. A., Narten, D. T., Nordmark, E. & Soliman, H. (2007). Neighbor Discovery for IP version 6 (IPv6). https://doi.org/10.17487/RFC4861

Wikipedia contributors. (2021). File:RIR Service Regions Map 2015.png — Wikipedia, The Free Encyclopedia [[Online; accessed 15-June-2022]]. https://en.wikipedia.org/w/index.php?title=File:RIR_Service_Regions_Map_2015.png&oldid=1057288877

Wikipedia contributors. (2022). IPv6 transition mechanism — Wikipedia, The Free Encyclopedia [[Online; accessed 16-June-2022]]. https://en.wikipedia.org/w/index.php?title=IPv6_transition_mechanism&oldid=1092642911

Anhang

Anhang A

Abbildungen

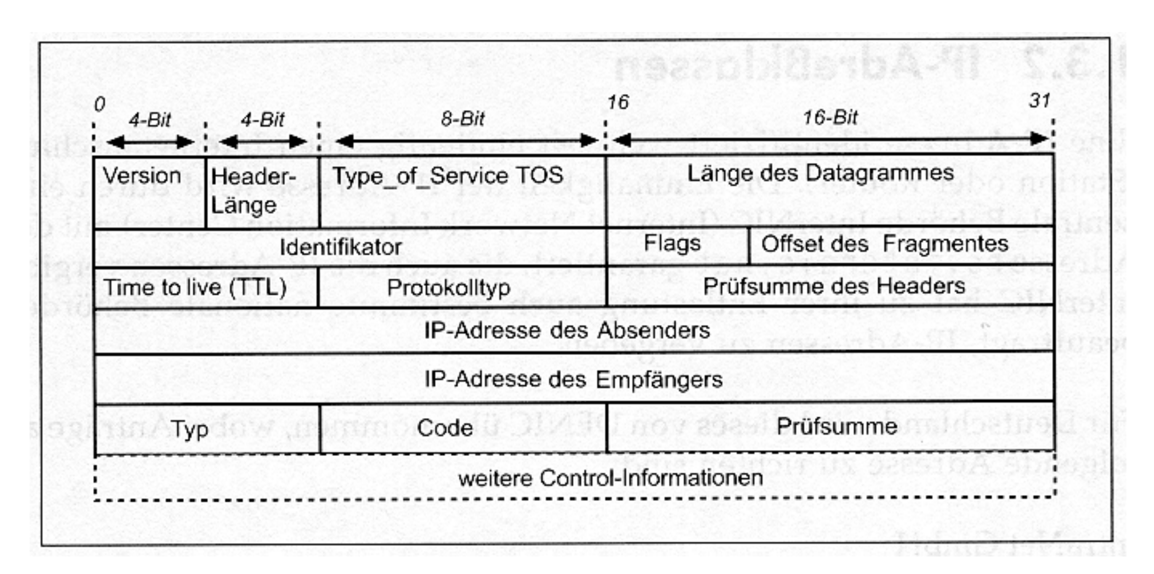


Abbildung A.1: IPv4-Header
Abbildung von Poiger (Poiger, 1998)

Anhang B

DNS-Lookups

Result for gar-vs.de/A without DNSSEC validation and without DNSSEC detail:


```
{
  "Status": 0 /* NOERROR */,
  "TC": false,
  "RD": true,
  "RA": true,
  "AD": false,
  "CD": true,
  "Question": [
    {
      "name": "gar-vs.de.",
      "type": 1 /* A */
    }
  ],
  "Answer": [
    {
      "name": "gar-vs.de.",
      "type": 1 /* A */,
      "TTL": 3600,
      "data": "185.237.65.171"
    }
  ],
  "Comment": "Response from 193.227.117.124."
}
```


You may also resolve directly at: https://dns.google/resolve?name=gar-vs.de&type=A&cd=true

```
[redacted]~$ dig A gar-vs.de

; <<>> DiG 9.10.6 <<>> A gar-vs.de
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 35356
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
;; QUESTION SECTION:
;gar-vs.de.                     IN      A

;; ANSWER SECTION:
gar-vs.de.              3405    IN      A       185.237.65.171

;; Query time: 550 msec
;; SERVER: 2001:7c0:f00:1fff:ffff:ffff:ffff:fe09#53(2001:7c0:f00:1fff:ffff:ffff:ffff:fe09)
;; WHEN: Wed Jun 22 10:37:05 CEST 2022
;; MSG SIZE  rcvd: 54
```

```
[redacted]:~$ host gar-vs.de
gar-vs.de has address 185.237.65.171
gar-vs.de mail is handled by 10 mx1.belwue.de.
```

Abbildung B.1: DNS-Lookup für gar-vs.de (A)
https://dns.google/resolve?name=gar-vs.de&type=A&cd=true

Result for gar-vs.de/AAAA without DNSSEC validation and without DNSSEC detail:


```
{
  "Status": 0 /* NOERROR */,
  "TC": false,
  "RD": true,
  "RA": true,
  "AD": false,
  "CD": true,
  "Question": [
    {
      "name": "gar-vs.de.",
      "type": 28 /* AAAA */
    }
  ],
  "Authority": [
    {
      "name": "gar-vs.de.",
      "type": 6 /* SOA */,
      "TTL": 1800,
      "data": "gar-vs.de. hostmaster.gar-vs.de. 2021090903 86400 7200 3600000 172800"
    }
  ],
  "Comment": "Response from 194.0.182.1."
}
```


You may also resolve directly at: https://dns.google/resolve?name=gar-vs.de&type=AAAA&cd=true

```
[redacted]~$ dig AAAA gar-vs.de

; <<>> DiG 9.10.6 <<>> AAAA gar-vs.de
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 48113
;; flags: qr rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
;; QUESTION SECTION:
;gar-vs.de.                     IN      AAAA

;; AUTHORITY SECTION:
gar-vs.de.              3600    IN      SOA     gar-vs.de. hostmaster.gar-vs.de. 2021090903 86400 7200 3600000 172800

;; Query time: 60 msec
;; SERVER: 2001:7c0:f00:1fff:ffff:ffff:ffff:fe09#53(2001:7c0:f00:1fff:ffff:ffff:ffff:fe09)
;; WHEN: Wed Jun 22 10:33:50 CEST 2022
;; MSG SIZE  rcvd: 85
```

```
[redacted]~$ host gar-vs.de
gar-vs.de has address 185.237.65.171
gar-vs.de mail is handled by 10 mx1.belwue.de.
```

Abbildung B.2: DNS-Lookup für gar-vs.de (AAAA)
https://dns.google/resolve?name=gar-vs.de&type=AAAA&cd=true

```
{
  "Status": 0 /* NOERROR */,
  "TC": false,
  "RD": true,
  "RA": true,
  "AD": false,
  "CD": false,
  "Question": [
    {
      "name": "hs-furtwangen.de.",
      "type": 1 /* A */
    }
  ],
  "Answer": [
    {
      "name": "hs-furtwangen.de.",
      "type": 1 /* A */,
      "TTL": 20727,
      "data": "141.28.2.12"
    }
  ]
}
```


```
[redacted]~$ dig A hs-furtwangen.de

; <<>> DiG 9.10.6 <<>> A hs-furtwangen.de
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 53981
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
;; QUESTION SECTION:
;hs-furtwangen.de.              IN      A

;; ANSWER SECTION:
hs-furtwangen.de.       80283   IN      A       141.28.2.12

;; Query time: 1 msec
;; SERVER: 2001:7c0:f00:1fff:ffff:ffff:ffff:fe09#53(2001:7c0:f00:1fff:ffff:ffff:ffff:fe09)
;; WHEN: Wed Jun 22 10:47:31 CEST 2022
;; MSG SIZE  rcvd: 61
```

```
[redacted]~$ host hs-furtwangen.de
hs-furtwangen.de has address 141.28.2.12
hs-furtwangen.de mail is handled by 50 mx.rz.hs-furtwangen.de.
```

Abbildung B.3: DNS-Lookup für gar-vs.de (A)
https://dns.google/resolve?name=gar-vs.de&type=A&cd=true

Result for hs-furtwangen.de/AAAA with DNSSEC validation and without DNSSEC detail:


```
{
  "Status": 0 /* NOERROR */,
  "TC": false,
  "RD": true,
  "RA": true,
  "AD": false,
  "CD": false,
  "Question": [
    {
      "name": "hs-furtwangen.de.",
      "type": 28 /* AAAA */
    }
  ],
  "Authority": [
    {
      "name": "hs-furtwangen.de.",
      "type": 6 /* SOA */,
      "TTL": 1586,
      "data": "ns.hs-furtwangen.de. dnsmaster.hs-furtwangen.de. 2130848949 10800 1800 1209600 38400"
    }
  ]
}
```


```
[redacted]~$ dig AAAA hs-furtwangen.de

; <<>> DiG 9.10.6 <<>> AAAA hs-furtwangen.de
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 33591
;; flags: qr rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
;; QUESTION SECTION:
;hs-furtwangen.de.              IN      AAAA

;; AUTHORITY SECTION:
hs-furtwangen.de.       4664    IN      SOA     ns.hs-furtwangen.de. dnsmaster.hs-furtwangen.de. 2130839325 10800 1800 1209600 38400

;; Query time: 1 msec
;; SERVER: 2001:7c0:f00:1fff:ffff:ffff:ffff:fe09#53(2001:7c0:f00:1fff:ffff:ffff:ffff:fe09)
;; WHEN: Wed Jun 22 10:47:49 CEST 2022
;; MSG SIZE  rcvd: 94
```

```
[redacted]~$ host hs-furtwangen.de
hs-furtwangen.de has address 141.28.2.12
hs-furtwangen.de mail is handled by 50 mx.rz.hs-furtwangen.de.
```

Abbildung B.4: DNS-Lookup für gar-vs.de (AAAA)
https://dns.google/resolve?name=gar-vs.de&type=AAAA&cd=true

Result for cloudflare.com/A without DNSSEC validation and without DNSSEC detail:


```
{
  "Status": 0 /* NOERROR */,
  "TC": false,
  "RD": true,
  "RA": true,
  "AD": false,
  "CD": true,
  "Question": [
    {
      "name": "cloudflare.com.",
      "type": 1 /* A */
    }
  ],
  "Answer": [
    {
      "name": "cloudflare.com.",
      "type": 1 /* A */,
      "TTL": 300,
      "data": "104.16.132.229"
    },
    {
      "name": "cloudflare.com.",
      "type": 1 /* A */,
      "TTL": 300,
      "data": "104.16.133.229"
    }
  ],
  "Comment": "Response from 162.159.6.6."
}
```


```
[redacted]~$ dig A cloudflare.com

; <<>> DiG 9.10.6 <<>> A cloudflare.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 55828
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
;; QUESTION SECTION:
;cloudflare.com.                        IN      A

;; ANSWER SECTION:
cloudflare.com.         295     IN      A       104.16.133.229
cloudflare.com.         295     IN      A       104.16.132.229

;; Query time: 1 msec
;; SERVER: 2001:7c0:f00:1fff:ffff:ffff:ffff:fe09#53(2001:7c0:f00:1fff:ffff:ffff:ffff:fe09)
;; WHEN: Wed Jun 22 10:37:37 CEST 2022
;; MSG SIZE  rcvd: 75
```

```
[redacted]~$ host cloudflare.com
cloudflare.com has address 104.16.133.229
cloudflare.com has address 104.16.132.229
cloudflare.com has IPv6 address 2606:4700::6810:85e5
cloudflare.com has IPv6 address 2606:4700::6810:84e5
cloudflare.com mail is handled by 10 mailstream-west.mxrecord.io.
cloudflare.com mail is handled by 20 mailstream-central.mxrecord.mx.
cloudflare.com mail is handled by 10 mailstream-east.mxrecord.io.
```

Abbildung B.5: DNS-Lookup für cloudflare.com (A)
https://dns.google/resolve?name=cloudflare.com&type=A&cd=true

Result for cloudflare.com/AAAA without DNSSEC validation and without DNSSEC detail:


```
{
  "Status": 0 /* NOERROR */,
  "TC": false,
  "RD": true,
  "RA": true,
  "AD": false,
  "CD": true,
  "Question": [
    {
      "name": "cloudflare.com.",
      "type": 28 /* AAAA */
    }
  ],
  "Answer": [
    {
      "name": "cloudflare.com.",
      "type": 28 /* AAAA */,
      "TTL": 300,
      "data": "2606:4700::6810:84e5"
    },
    {
      "name": "cloudflare.com.",
      "type": 28 /* AAAA */,
      "TTL": 300,
      "data": "2606:4700::6810:85e5"
    }
  ],
  "Comment": "Response from 162.159.8.55."
}
```


```
[redacted]~$ dig AAAA cloudflare.com

; <<>> DiG 9.10.6 <<>> AAAA cloudflare.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 1201
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
;; QUESTION SECTION:
;cloudflare.com.                        IN      AAAA

;; ANSWER SECTION:
cloudflare.com.         300     IN      AAAA    2606:4700::6810:85e5
cloudflare.com.         300     IN      AAAA    2606:4700::6810:84e5

;; Query time: 18 msec
;; SERVER: 2001:7c0:f00:1fff:ffff:ffff:ffff:fe09#53(2001:7c0:f00:1fff:ffff:ffff:ffff:fe09)
;; WHEN: Wed Jun 22 10:33:16 CEST 2022
;; MSG SIZE  rcvd: 99
```

```
[redacted]~$ host cloudflare.com
cloudflare.com has address 104.16.133.229
cloudflare.com has address 104.16.132.229
cloudflare.com has IPv6 address 2606:4700::6810:85e5
cloudflare.com has IPv6 address 2606:4700::6810:84e5
cloudflare.com mail is handled by 10 mailstream-west.mxrecord.io.
cloudflare.com mail is handled by 20 mailstream-central.mxrecord.mx.
cloudflare.com mail is handled by 10 mailstream-east.mxrecord.io.
```

Abbildung B.6: DNS-Lookup für cloudflare.com (AAAA)
https://dns.google/resolve?name=cloudflare.com&type=AAAA&cd=true

Anhang C

Acknowledgments

Diese Personen/Organisationen/etc. haben nicht zwingend etwas mit dem Inhalt dieses Dokumentes zu tun. Teilweise mögen Sie nur die Software/Vorlagen gemacht haben, mit welchen dieses Dokument erstellt wurde. Das möchte ich hiermit anerkennen.

C.1 Vorlage für dieses Dokument

- Fabian Berner: http://webuser.hs-furtwangen.de/~berner/wa.html (This Link is dead)
- Manuel Kühner: https://bedienhaptik.de/latex-template/
- lbausch (MIT-License): https://github.com/lbausch/latex-template-hfu